H5O_GET_INFO_BY_IDX1

Expand all Collapse all

- Jump to ...
- Summary
- Description
- Example
- Switch language ...
- C
- C++
- FORTRAN
- JAVA

H5O_GET_INFO_BY_IDX1

Retrieves the metadata for an object, identifying the object by an index position

> *As of HDF5-1.12 this function is deprecated in favor of the function H5O_GET_INFO_BY_IDX2 or the macro H5O_GET_INFO_BY_IDX .*

Procedure:

H5O_GET_INFO_BY_IDX1(loc_id, group_name, index_field, order, n, object_info, lapl_id)

Signature:

```
herr_t H5Oget_info_by_idx1( hid_t loc_id, const char *group_name, H5_index_t index_field, H5_iter_order_t
order, hsize_t n, H5O_info_t *object_info, hid_t lapl_id )
```

```
SUBROUTINE h5oget_info_by_idx_f(loc_id, group_name, index_field, order, n, &
       object_info, hdferr, lapl_id)

    USE, INTRINSIC :: ISO_C_BINDING
    IMPLICIT NONE
    INTEGER(HID_T)  , INTENT(IN)            :: loc_id
    CHARACTER(LEN=*), INTENT(IN)            :: group_name
    INTEGER         , INTENT(IN)            :: index_field
    INTEGER         , INTENT(IN)            :: order
    INTEGER(HSIZE_T), INTENT(IN)            :: n
    TYPE(h5o_info_t), INTENT(OUT), TARGET   :: object_info
    INTEGER         , INTENT(OUT)           :: hdferr
    INTEGER(HID_T)  , INTENT(IN) , OPTIONAL :: lapl_id
```

Related Fortran2003 Derived Type: h5o_info_t

```
TYPE, BIND(C) :: space_t
     INTEGER(hsize_t) :: total ! Total space for storing object header in file
     INTEGER(hsize_t) :: meta  ! Space within header for object header metadata
                               ! information
     INTEGER(hsize_t) :: mesg  ! Space within header for actual message
                               ! information
     INTEGER(hsize_t) :: free  ! Free space within object header
  END TYPE space_t

  TYPE, BIND(C) :: mesg_t
     INTEGER(c_int64_t) :: present ! Flags to indicate presence of message type
                                   ! in header
     INTEGER(c_int64_t) :: shared  ! Flags to indicate message type is shared
                                   ! in header
  END TYPE mesg_t

  TYPE, BIND(C) :: hdr_t
     INTEGER :: version ! Version number of header format in file
     INTEGER :: nmesgs  ! Number of object header messages
     INTEGER :: nchunks ! Number of object header chunks
     INTEGER :: flags   ! Object header status flags
     TYPE(space_t)  :: space
     TYPE(mesg_t)   :: mesg
  END TYPE hdr_t

  ! Extra metadata storage for obj & attributes
  TYPE, BIND(C) :: H5_ih_info_t
     INTEGER(hsize_t) :: index_size ! btree and/or list
     INTEGER(hsize_t) :: heap_size
  END TYPE H5_ih_info_t

  TYPE, BIND(C) :: meta_size_t
     TYPE(H5_ih_info_t) :: obj  ! v1/v2 B-tree & local/fractal heap for
                                ! groups, B-tree for chunked datasets
     TYPE(H5_ih_info_t) :: attr ! v2 B-tree & heap for attributes
  ENDTYPE meta_size_t

  TYPE, BIND(C) :: h5o_info_t
     INTEGER(C_LONG)  :: fileno     ! File number that object is located in
     INTEGER(haddr_t) :: addr       ! Object address in file
     INTEGER(C_INT)   :: type       ! Basic object type (group, dataset, etc.)
     INTEGER          :: rc         ! Reference count of object

     INTEGER, DIMENSION(8) :: atime ! Access time        !    -- NOTE --
     INTEGER, DIMENSION(8) :: mtime ! Modification time  ! Returns an integer
     INTEGER, DIMENSION(8) :: ctime ! Change time        ! array as specified
     INTEGER, DIMENSION(8) :: btime ! Birth time         ! in Fortran intrinsic
                                                         ! DATE_AND_TIME(VALUES)

     INTEGER(hsize_t) :: num_attrs  ! # of attributes attached to object

     TYPE(hdr_t) :: hdr

     TYPE(meta_size_t) :: meta_size
  END TYPE h5o_info_t
```

Parameters:

hid_t `loc_id`	IN: Location identifier of object; may be a file, group, dataset, named datatype or attribute identifier
const char *`group_name`	IN: Name of group in which object is located
H5_index_t `index_field`	IN: Index or field that determines the order

H5_iter_order_t `order`	IN: Order within field or index
hsize_t n	IN: Object for which information is to be returned
H5O_info_t *`object_info`	OUT: Buffer in which to return object information
hid_t `lapl_id`	IN: Link access property list *—(Not currently used; pass as `NULL`.)*

Description:

H5O_GET_INFO_BY_IDX1 retrieves the metadata describing an object in the struct `object_info`, as specified by the location, `loc_id`, group name, `group_name`, the index by which objects in that group are tracked, `index_field`, the order by which the index is to be traversed, `order`, and an object's position n within that index .

`object_info`, in which the object information is returned, is a struct of type *H5O_info_t*. This struct type is described in the H5O_GET_INFO fun ction entry.

If `loc_id` fully specifies the group in which the object resides, `group_name` can be a dot (.).

The link access property list, `lapl_id`, is not currently used; it should be passed in as `NULL`.

Returns:

Returns a non-negative value if successful; otherwise returns a negative value.

Example:

History:

Release	Change
1.12.0	Function was deprecated.
1.10.3	Function renamed to `H5Oget_info_by_idx1`.
1.8.11	Fortran subroutine introduced in this release.
1.8.0	Function introduced in this release.

--- Last Modified: September 05, 2019 | 01:11 PM